HB-MP$^+$ Protocol: An Improvement on the HB-MP Protocol

Xuefei Leng, Keith Mayes, Konstantinos Markantonakis
Smart Card Centre, Information Security Group,
Department of Mathematics, Royal Holloway, University of London
x.f.leng@rhul.ac.uk
Keith.Mayes@rhul.ac.uk
K.Markantonakis@rhul.ac.uk

***Abstract*—In this paper, we propose an enhanced version of the HB-MP authentication protocol, called the HB-MP$^+$ protocol. The HB-MP protocol is a lightweight authentication protocol that is suitable for use in passive radio frequency identification (RFID) systems. The HB-MP$^+$ protocol overcomes the man-in-the-middle attack to which the basic HB-MP protocol is vulnerable while maintaining its suitability to low-cost passive RFID systems. We show an effective man-in-the-middle attack against the HB-MP protocol where the attacker utilizes the predictable rotation of the secret key. We enhance the HB-MP protocol by randomizing the rotation of the secret key, which eliminates the vulnerability. We also propose the use of round keys that may be produced by rotation or, more generally, by a one-way function. We analyse the security and performance improvements of our HB-MP$^+$ protocol and find it to be suitable for passive RFID systems.**

I. Introduction

Radio frequency identification (RFID) systems have generated great interest in recent years and RFID tags are poised to replace barcodes in the near future. Since the cost of RFID tags is among the biggest impediments to widespread use, the computing power and resource of RFID tags are expected to remain extremely limited. Hence the algorithms and protocols used in RFID systems are required to be extremely efficient. However, the security features needed in RFID systems are almost the same as needed in other systems: authenticity, integrity, confidentiality, untraceability and availability.

In 2001, Hopper and Blum [3] proposed an authentication protocol to meet the demand of human authentication, known as the HB protocol. The HB protocol relies on the computation hardness of the Learning Parity with Noise (LPN) problem, and uses only dot products of binary vectors and a random noise bit, so it is very lightweight. In 2005, Juels and Weis [4] adopted this HB protocol into RFID systems because of the similarity between human and tags. The authors also presented an active attack on the HB protocol, and they proposed an enhanced version called the HB$^+$ protocol. Later in the same year, Katz and Shin [9] proved the parallel concurrent security property of the HB and HB$^+$ protocols, but their proofs only imply meaningful security for $\varepsilon < 1/4$. Follow on work done by Katz and Smith [10] extend the proofs of security for the full HB and HB$^+$ protocols for arbitrary $\varepsilon < 1/2$. Unfortunately These two protocols were shown by Gilbert et al. [5] to be vulnerable to certain man-in-the-middle attacks. In 2006, Bringer et al. [11] proposed an enhancement of HB$^+$ called HB^{++}. Piramuthu [7] had a survey of the HB-family protocols and proposed another modification of the HB$^+$ protocol, in which the bit-wise rotations are varied for each round and the message flow is simplified (saving one bit per round). In 2007, Duc and Kim presented a variation of HB$^+$ protocol called HB*, which is resistant to Gilbert et al.'s attack [12], But Piramuthu [8] broke HB* and proposed his modified protocol. In 2008, Gilbert et al. [6] proposed their thorough analysis on the HB protocol families and proposed two new protocols called *RANDOM*-HB$^\#$ and HB$^\#$: *RANDOM*-HB$^\#$ avoids many practical drawbacks of HB$^+$, remains provably resistant to attacks in the model of Juels and Weis [4], and is also provably resistant to a broader class of active attacks. However, *RANDOM*-HB$^\#$ is required to store two random matrices, which make the storage costs insurmountable to the tags. HB$^\#$ enhanced *RANDOM*-HB$^\#$ by using Toeplitz matrices [13][14] to improve the performance. At the time when this paper is written, the latest paper on improvement of HB$^+$ is the "Trusted-HB" protocol, which uses a LFSR-based Toeplitz hashing [13] to enhance the security of HB$^+$.

In early 2007, Munilla and Peinado [1] proposed a prominent protocol called HB-MP. The HB-MP protocol has a fresh way of exchanging messages and improved attack resistance whilst retaining the simplicity of the HB family. However, the protocol is still vulnerable and an improved countermeasure is discussed within this paper.

The paper is organised as follows. In Section II, the LNP problem and the HB$^+$ Protocol are introduced and analysed, this prepares the discussion in subsequent sections. In Section III, we introduce an early step of the HB-MP protocol and a man-in-the-middle attack that effectively breaks it. Section IV introduces the HB-MP protocol which had some measures added to defend against this attack. It furthermore describes the vulnerability of HB-MP protocol which could enable the attack mentioned in section III. Section V proposes an improved HB-MP protocol to resist such attack. Section VI proposes a general form of HB-MP$^+$ protocol. Section VII gives security and performance analysis of the HB-MP$^+$ protocol.

II. LPN Problems and HB$^+$ Protocol

A. LPN Problems

All the protocols of the HB family are based on the conjectured hardness of the Learning Parity in the Presence of Noise, or LPN problem. Here we offer the definition of the LPN problem:

> **LPN Problem:** The LPN problem with security parameters q, k, η with $\eta \in (0, 1/2)$ is defined as follows: given a random $q \times k$ binary matrix A, a random k-bit vector x, a vector v such that $|v| \leq \eta q$, and the product $z = A \cdot x \oplus v$, find a k-bit vector x' such that $|A \cdot x' \oplus z| \leq \eta q$, where $|v|$ denotes the Hamming weight of vector v.

The LPN problem is known to be NP-Hard [2]; currently no polynomial algorithm is known to solve the LPN problem. Hopper and Blum [3] and Juels and Weis [4] cited the BKW algorithm which is considered to be fastest in solving LPN problem. However in Gilbert et al. [6], the authors used the results from other researchers, conclude the former way of defining security parameters of LPN problem needs adjustment, as the BKW algorithm is improved significantly. For example, it was thought that a LPN problem has the length of the secret $k = 224$ and the noise level $\eta = 0.25$ could achieve around 80-bit security. Unfortunately Gilbert et al. [6] cited new research showing that, using the new BKW algorithm, $k = 224$ and $\eta = 0.25$ can only offer a security level no more than 52 bit. They conservatively proposed that $k = 512$ and $\eta = 0.25$ should provide a good security level.

B. HB$^+$ Protocol

Since Juels and Weis [4] introduced the HB$^+$ Protocol, considerable research interests were generated and several protocols based on this protocols were introduced in the previous section. Since the HB-MP protocol is also inspired by HB$^+$ Protocol, it is necessary to have a brief introduction to this protocol. Some notation will aslo be introduced which are consistent with that used in Munilla and Peinado's paper [1].

k	length of the secret keys shared by the reader and the tag.
x, y	k bits secret keys shared by the reader and the tag.
a, b	random k-bits binary vectors.
v	noise bit; $v = 1$ with probability $\eta \in [0, 1/2]$.
$\oplus$	XOR operation.
$a \cdot x$	scalar product of vector a and x
q	Number of rounds in an authentication session

TABLE I
Notations for HB$^+$ Protocol

Step 1. The tag chooses at random a k-bit binary vector b, and sends it to the reader.

Step 2. The reader generates at random a challenge a and send it to the tag.

Step 3. The tag computes $z = a \cdot x \oplus b \cdot y \oplus v$ and sends z.

Step 4. The reader checks whether $z = a \cdot x \oplus b \cdot y$.

The HB$^+$ protocol runs q rounds and Fig.1 shows one round. If the non-match rounds exceeds a threshold t, the tag is rejected and otherwise it will be accepted. People reading this protocol would naturally consider that the error reject rate can be fairly high and it is possible for a very lucky fake tag to be successfully authenticated. From probability theory, one can give the equations of false rejection rate P_{FR}, and false acceptance P_{FA}:

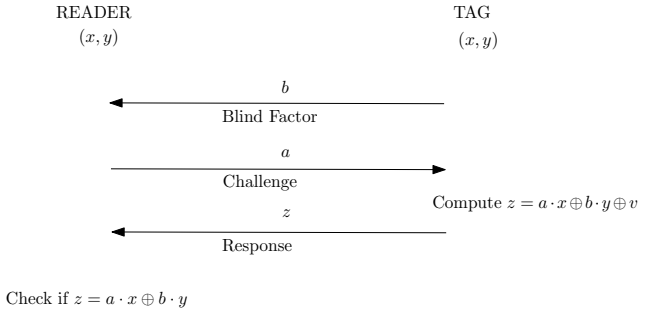

Fig. 1. A Round of HB$^+$ Protocol

$$P_{FR} = \sum_{i=t+1}^{q} \binom{q}{i} \eta^i (1-\eta)^{q-i} \tag{1}$$

$$P_{FA} = \sum_{i=0}^{t} \binom{q}{i} 2^{-q} \tag{2}$$

It is clear from the equations that both P_{FR} and P_{FA} are irrelevant to the lengths of the secret k, they are only relevant to q, t and η. In the original HB$^+$ protocol, a threshold of $t = \eta q$ is suggested. However in Gilbert et al. [6], a table describing the relations of security parameters and error rates shows the default choice $t = \eta q$ gives an unacceptably high false rejection rate. For example, when $q = 60$, $\eta = 0.25$, $k = 224$, P_{FR} can be as large as 0.43! It is hard to imagine any practical scenario where a probability higher than 1% of rejecting a legitimate tag could be tolerated.

Another obvious problem of implementing the HB$^+$ protocol is the transmission costs of the q rounds communication. Actually considering the high false rejection rate, a genuine tag might need to run over q rounds to get authenticated. In each round, a pack of three k-bit messages have to be transmitted. Gilbert et al. [6] gave some description of the transmission cost which shows the transmission cost of HB$^+$ protocol with previously proposed security parameters, $q = 60$, $\eta = 0.25$, $k = 224$, will have to transmit at least 26,984 bits of data just for a whole (q rounds) authentication. For the more secure HB$^+$ protocol with $k = 512$, there needs at least 61,500 bits of data to be transmitted. All this calculation is not including other necessary transmission overloads like the transmission time intervals and error-checking code attached. So HB$^+$ protocol is still impractical for current RFID systems.

III. The HB-MP$'$ Protocol and Its Weakness

A. The HB-MP$'$ Protocol

HB-MP protocol is a variation of HB$^+$ protocol. There is an important step in developing the HB-MP protocol from

the HB$^+$ protocol called the HB-MP$'$ protocol. It is a direct modification of HB$^+$ but vulnerable to the man-in-middle attack proposed by Gilbert et al. [5]. The HB-MP protocol is an enhanced version of HB-MP$'$ to resist such attack.

The protocol of HB-MP$'$ is composed of q rounds. One of which is depicted in Fig.2 and described as follows:

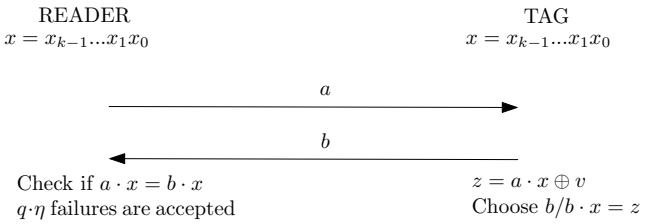

Fig. 2. A Round of HB-MP′ Protocol

Step 1. The reader chooses at random a k-bit binary vector a, and sends it to the tag.

Step 2. The tag computes z as follows: $z = a \cdot x \oplus v$ and looks for a k-bit binary vector b such that $b \cdot x = z$.

Step 3. The tag sends b to the reader.

Step 4. The reader checks whether $b \cdot x = a \cdot x$.

Munilla and Peinado [1] give a detailed explanation about finding x knowing the vectors a and b is at least as difficult as solving the LPN problem. The essential change made by the HB-MP′ protocol is that the tag side takes the computation load of picking the response b. In their paper, they give a neat algorithm on picking b when $\eta = 0.25$:

Algorithm 1. Input: a, x. Output: b such that $b \cdot x = a \cdot x \oplus v$, where $v = 1$ with probability $1/4$.

```
    Computes z = a · x
    Generates at random k-bit binary
vector b
    If b · x = z
       Sends b
    else
       Generates and sends a new random
k-bit vector b
    end
```

From the algorithm, one can know that the possibility that $b \cdot x \neq a \cdot x$ is $0.5 \times 0.5 = 0.25$, that means $\eta = 0.25$. If the b is checked n times before it is sent, then $\eta = 1/2^{n+1}$ In Algorithm 2, we give a general form of n times checking on b before sending it.

Algorithm 2. Input: a, x, n. Output: b such that $b \cdot x = a \cdot x \oplus v$, where $v = 1$ with probability $1/2^{n+1}$.

```
    Computes z = a · x
    While n ≥ 1
       Generates at random k-bit vector b
       If b · x = z
         Break
       n = n − 1
    end While
    Sends b
    end
```

$n = 1$, $\eta = 0.25$ and $n = 2$, $\eta = 0.125$ are both practical and popular choices.

B. A Man-in-the middle Attack on the HB-MP$'$ Protocol

HB-MP$'$ is the prototype of HB-MP protocol but it is vulnerable to a man-in-the-middle attack similar to the one proposed by Gilbert et al. [5]. Munilla and Peinado [1] mentioned this type of attack on the HB-MP$'$ protocol, however, the attack was not described within their paper. Here we give an example of such an attack. It is reasonably assumed that

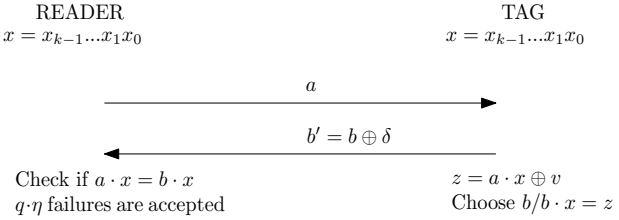

Fig. 3. A Successful Active Attack against HB-MP′ Protocol

the adversary is capable of manipulating challenges sent by a legitimate tag to a legitimate reader during the authentication procedure, checking whether this manipulation results (or not) in an authentication failure. The attack is illustrated in Fig.3 onto a single round of the HB$^+$ protocol. The attacker chooses a constant k-bit vector δ and use it to perturb the response sent by a legitimate tag to the legitimate reader: $b' = b \oplus \delta$ for each of the q rounds of authentication. If the authentication process is successful, then it must be true that $\delta \cdot x = 0$ with overwhelming probability. If authentication doesn't succeed then $\delta \cdot x = 1$ with overwhelming probability.

We use the same δ in all q rounds of the protocol. Acceptance or rejection by the reader would reveal one bit of secret information x. To retrieve the k-bit secret x, it is enough to repeat the full protocol k times for linearly independent δ 's, and to solve the resulting system. Conveniently, the attacker can choose δs with a single non-zero bit and this non-zero bit is different for each δ. Once x has been derived, the attacker is able to impersonate the tag. Another side effect of the disclosure of x is that the privacy of the tag's identity is also compromised.

IV. The HB-MP Protocol and Its Weakness

A. The HB-MP Protocol

The HB-MP protocol is an enhancement of HB-MP$'$. With the same notation of HB-MP$'$, there are some more notations:

The protocol also runs q rounds to achieve one authentication, one of which, the ith round, is depicted in Fig.4 and described as follows:

Step 1. The reader chooses at random an m-bit binary vector a and sends it to the tag.

m	length of the message exchanged between the parties.
x, y	k bits secret keys shared by the reader and the tag.
xm	the m-bit binary vector consisting of the m least significant bits of x.
a, b	a, b are m-bit long in the following protocols
$Rot(p, u)$	the bitwise left rotate operator. The operand p is rotated u positions.

TABLE II
MORE NOTATIONS FOR HB-MP PROTOCOL

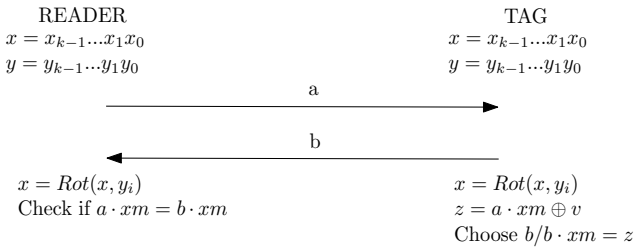

Fig. 4. The ith Round of HB-MP Protocol

Step 2. The tag computes $x = Rot(x, y_i)$, where y_i is the ith bit of the key y.

Step 3. The tag computes z as follows:

$$z = a \cdot xm \oplus v \tag{3}$$

and looks for a k-bit binary vector b such that $b \cdot xm = z$.

Step 4. The tag sends b to the reader.

Step 5. The reader computes the x in the ith round as $x = Rot(x, y_i)$, where y_i is the ith bit of y.

Step 6. The reader checks if

$$a \cdot xm = b \cdot xm \tag{4}$$

After q rounds, the reader trusts the tag is legitimate if the failures are below $q \cdot \eta$ rounds.

B. A Man-in-the-middle Attack on the HB-MP Protocol

Defending against the man-in-the-middle attack as proposed by Gilbert et al. has been considered in the HB-MP protocol, hence the rotation of xm. But this rotation has its own weakness. In the design of the HB-MP protocol, for every new session, xm needs to be identical in the ith round. It is not stated clearly about when to start and end an authentication session. It is reasonable to suppose that when the tag enters the electromagnetic field and starts to talk with the reader, an authentication session begins and when the q-round enquiry is finished or the tag departs from the electromagnetic field of the reader, the authentication session ends. Since $x = Rot(x, y_i)$, so xm in the first round of all the authentication sessions should be the same. If the attacker pretends to be a valid reader, he can initiate repetitive authentication sessions, initially restricted to the first round. The techniques used in last section can then be exploited to reveal the tag's first round xm. If the attacker observes the ith round, he is able to reveal the xm used in the ith round.

The reason why the protocol has to use the same xm between authentication sessions is the synchronisation problem. If the value of x is updated to the rotated value after every authentication session on both the reader and tag side, a new reader will not be able to verify the updated tag and a valid new tag can not be verified by the updated reader, since the values of x stored in the reader and tags are not the same. Unless all the readers and tags are updated at the same time after every authentication session, which is expensive and technically difficult, the synchronization problem forbids the HB-MP protocols to change the xm.

Even if the synchronization problem is solved, the x is updated in every authentication session. There is still a way to conduct the man-in-the-middle attack. The length of x and y is k, if in an authentication session, the protocol runs k rounds, the x will be rotated p bits, here p is the number of '1' in y, so if the attacker runs the protocol for k times, namely k^2 rounds, the x will be rotated $p \cdot k$ times and it is rotated back to its initial value. so a repeat of xm happens again. Since the proposed x is 512 bits, so 262,144 rounds will definitely generate a repeated xm. It is an affordable attack.

V. AN IMPROVED HB-MP PROTOCOL

The vulnerability of this protocol stems from the predictable repetition of xm, if the rotation is random in each round, the repetition of xm is unpredictable, thus the attack is defended.

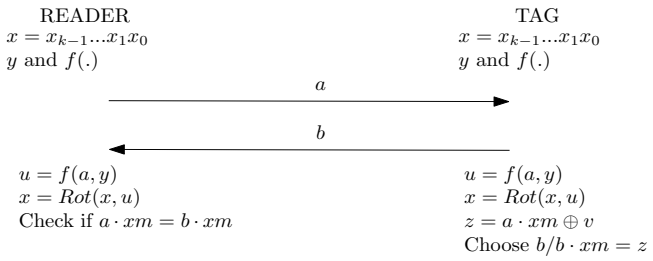

Fig. 5. The ith Round of the Improved HB-MP Protocol

The notations are same with the original HB-MP protocol except a one way function $f(.)$ and an intermediate value $u = f(a, y)$. Instead of using the default threshold ηq, we define the threshold t. Because Gilbert et al. [6] cites the result of other research, describing threshold ηq can cause unreasonably high false rejection rate. The threshold t can be adjusted to a value larger than ηq to reduce the false rejection rate, and retain a low false acceptance rate at the same time. This improved protocol also has q rounds and if the failures do not pass the threshold t, the tag is authenticated successfully. The ith round is depicted in Fig.5 and described as follows:

Step 1. The reader chooses at random a m-bit binary vector a and send it to the tag.

Step 2. The reader and tag compute $u = f(a, y)$ and $x = Rot(x, u)$. xm is selected as the first m-bits of current x.

Step 3. The tag computes z as follows:

$$z = a \cdot xm \oplus v \tag{5}$$

and looks for a m-bit binary vector b such that $b \cdot xm = z$.

Step 4. The tag sends b to the reader.

Step 5. The reader computes the x in the ith round as $x = Rot(x, t)$ and selects xm from this x.

Step 6. The reader checks if

$$a \cdot xm = b \cdot xm \tag{6}$$

By using the random number a, The improved protocol makes the rotation of x unpredictable. An advantage of this improvement is that the original form of the HB-MP protocol is kept and the only change is the computation operated inside the tag and reader. This protocol improves the HB-MP protocol by making the rotation of the secret unpredictable to the attacker.

VI. An Abstract Form of HB-MP$^+$ Protocol

The core idea of the improved protocol is to use some additional random bits generated by the reader to randomize the rotation. The evolutionary design idea from HB-MP′ to HB-MP was to rotate the secret key x in each round. The improved HB-MP protocol makes the rotation unpredictable by adding randomness. If we extend the design idea a step further, the essential part of defending against man-in-the-middle attack is to use a random secret in each round, namely a **Round Key**. The vulnerability of HB-MP comes from the predictability of the round key. So if the focus of protocol design points to the generation of a round key by using the random bits and shared secrets, a new protocol can be proposed. The HB-MP$^+$ protocol we proposed is called in abstract form because the one-way function $f(.)$ is not concrete. HB-MP$^+$ is also in accordance of the naming convention of HB-family protocols. The HB-MP$^+$ protocol also has adjustable threshold t to improve the false rejection rate. Fig. 6 shows one round of the HB-MP$^+$ protocol. *Step 1.* The reader chooses at random

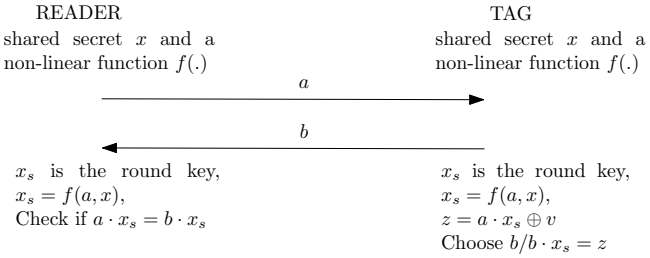

Fig. 6. The ith Round of the Abstract HB-MP$^+$ Protocol

a m-bit binary vector a and send it to the tag.

Step 2. The reader and tag compute the round key $x_s = f(a, x)$. $f(.)$ is the one-way function.

Step 3. The tag computes z as follows:

$$z = a \cdot x_s \oplus v \tag{7}$$

and looks for a m-bit binary vector b such that $b \cdot x_s = z$.

Step 4. The tag sends b to the reader.

Step 5. The reader computes the $x_s = f(a, x)$, using the secret x and random number a.

Step 6. The reader checks if

$$a \cdot x_s = b \cdot x_s \tag{8}$$

The notations are almost the same with the improved HB-MP protocol, except that the non-linear function $f(.)$ is the abstraction from $Rot(.)$. Since rotation is a linear operation, the output of $f(.)$ should be less predictable. Using the bit operations, it is easy to implement a low-cost non-linear function $f(.)$. As $f(.)$ does not necessarily use rotation, the bits of x are not mentioned. The round key x_s is generated by a random number a and shared secret x. There is no need for another shared secret y and because the x is not changed, there are no synchronisation problems between the readers and tags.

VII. Security and Performance Analysis of The HB-MP$^+$ Protocol

A. Security Analysis

The improved HB-MP protocol and HB-MP$^+$ protocol is based on the HB-MP protocol. Since the way of transmitting messages is the same with the HB-MP protocol. The analysis in Munilla and Peinado [1] also applies to the HB-MP$^+$ protocol: A passive attacker has to solve the LPN problem to reveal the secret of tags(x, more specifically).

In the design of these two protocols, it is very important that the round key is calculated by random challenge a and the secret x. Since a is randomly generated by the reader and stored in the reader, the attacker cannot either predict a or modify a. If an attacker modifies a to a' to cheat the tag, the tag side will use a' to generate the round key x_s, which is not the same round key generated at the reader side since the reader still uses its own a. Thus the attacker cannot get any valuable response from the tag.

It is necessary to define the attacker's aim when we start to talk about the attack. In the two protocols we propose, the tag does not authenticate the reader. In other word, the tag does not care who is challenging it, it just gives back responses according to the challenge. So the attacker's meaningful aim is to fool the reader to authenticate a fake tag or to reveal the secret x. Man-in-the-middle attack presented in Gilbert et al. [5] can reveal the secret of the target protocols. To illustrate how our protocols defend against the man-in-the-middle attack, We give a model describing the ability of an active attacker. Let $\mathcal{A}$ be an adversary, who can intercept the communication between the reader and the tag. $\mathcal{A}$ can also pretend a reader to challenge the tag. $\mathcal{A}$ can block and modify the messages sent both by the reader and the tags. Finally $\mathcal{A}$ can fool the reader to answer multiply responses for a single challenge in each round (which is unlikely to happen to a reader with the predefined procedure to handle the information received). We only illustrate the attack to the HB-MP$^+$ protocol, which is the mature protocol in this paper. Fig.7 shows an assumed man-in-the-middle attack launched by the the attacker $\mathcal{A}$ against the HB-MP$^+$ protocol.

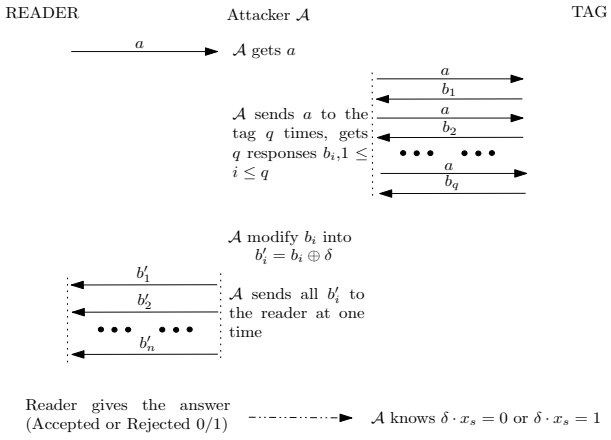

Fig. 7. Assumed Man-in-the-middle Attack on HB-MP$^+$ Protocol

If $\mathcal{A}$ is the man-in-the-middle and attacking the HB-MP$^+$ protocol, $\mathcal{A}$ gets the challenge a sent by an authentic reader, then $\mathcal{A}$ pretends to be a reader and challenge the tag with the same a q times. $\mathcal{A}$ receives q responses b_i $(1 \leq i \leq q)$ from the tag, $\mathcal{A}$ modified the responses into $b'_i = b_i \oplus \delta$ $(1 \leq i \leq q)$ and gives them back to the reader at one time. $\mathcal{A}$ gets the answer(accepted or rejected) from the reader. Then $\mathcal{A}$ knows that $x_s \cdot \delta = 1$ or $x_s \cdot \delta = 0$ with high probability, $\mathcal{A}$ can retrieve a bit of the round-key x_s generated by a. If the attacker can fool the reader to answer the challenge a continuously, the attacker can even reveal the whole x_s. However the $f(.)$ is a good one-way function, $\mathcal{A}$ cannot retrieve x from x_s and a. Since reader is generating challenge a randomly, which makes $\mathcal{A}$ unable to predict a, we can conclude that $\mathcal{A}$ can only retrieve the previous round keys but not the secret x (since $f(.)$ is a good one-way function). For a fresh challenge a, $\mathcal{A}$ cannot give the correct response back to fool the reader. So the HB-MP$^+$ protocol is secure against the active attacks similar to the one in [5].

For the improved HB-MP protocols presented in this paper, if $\mathcal{A}$ can derive enough rotations of x, $\mathcal{A}$ is able to find x at the end. Fortunately, in reality this assumed attack is unlikely to happen, the communication between RFID and the reader is predefined and sequenced. A normal reader will not tolerate more than one responses for a single challenge. State machine inside the reader will also start a fresh challenge after a q-round authentication session. To response one challenge so many times is the assumed condition to facilitate the attack. Another attack needs to be noticed is that the adversary can always response $b = a$. The reader needs to check that the response should be different from the challenge each round.

B. Performance Analysis

Because the HB-MP$^+$ protocol uses round keys, the secret is updated in each round, so m (the size of the round key) does not need to be as large as suggested in Gilbert et al. [6]. In their paper the authors suggested that $k = 512$ and $\eta = 0.25$ is good enough. In HB-MP$^+$ protocol, the secret x can be 512 bits while m can be significantly smaller. $m = 224$ offers a security level of 52-bit [6], which should be enough for the round key.

Another significant reduce of transmission cost comes from the fact that HB-MP$^+$ protocol transmits two messages instead of three each round. This will cut 1/3 transmission cost comparing to the HB$^+$ protocol.

Despite the improvement on the performance, HB-MP$^+$ protocol still suffers the same performance penalties with the HB$^+$ protocol. It still needs to run many rounds and there are still too much data needs to be transmitted for an authentication session. The transmission cost is too high for current RFID systems.

VIII. Conclusion

In this paper a vulnerability of the HB-MP protocol that may enable a successful man-in-the-middle attack has been identified, An enhanced version of HB-MP protocol is proposed that eliminates the vulnerability and keeps the simplicity of the original protocol. An abstract form of the HB-MP$^+$ protocol introduces the idea of random round keys is also proposed. This paper also improves the algorithms of picking random responses and gives adjustable threshold to reduce the false rejection rates. The abstract form of the HB-MP$^+$ protocol requires concrete the one-way function $f(.)$, however this is thought within the capabilities of RFID devices and comparable with the $Rot(.)$ function used in HB-MP.

References

[1] J.Munilla, A.Peinado. HP-MP: A Further Step in the HB-family of Lightweight authentication protocols. Computer Networks, 51(2007), pp.2262-2267

[2] E. R. Berlekamp, R. J. McEliece and V. Tilborg. On the Inherent Intractability of Certain Coding Problems. IEEE Transactions on Information Theory 24 (1978), 384C386.

[3] N.J.Hopper and M.Blum. Secure Human Identification Protocols. Advanced in Cryptology-ASIACRYPT'2001, Lecture Notes in Computer Science,Vol. 2248 Springer,2001,pp.52-66.

[4] A. Juels, S.Weis, Authenticating Pervasive Devices with Human Protocols, Advances in Cryptology-Crypto2005, Lecture Notes in Computer Science,Vol.3621, Springer, 2005, pp.293-308.

[5] H.Gilbert, M.Robshaw, H.Silbert, an Active Attack against HB$^+$-a Provable Secure Lightweighted Authentication Protocol. Cryptology ePrint Archive, Report 2005/237, 2005, http://eprint.iacr.org.

[6] H.Gilbert, M.Robshaw and Yannick Seurin, HB$^\#$: Increasing the Security and Efficiency of HB$^+$, 2008, To be published on EUROCRYPT 2008, Available from http://eprint.iacr.org.

[7] S.Piramuthu, HB and Related Lightweight Authentication Protocols for Secure RFID Tag/Reader Authentication, CollECTeR Europe Conference, Basel, Switzerland, 9-10 June, 2006.

[8] S. Piramuthu and Y. Tu, Modified HB Authentication Protocol, Western European Workshop on Research in Cryptology, July, Germany, 2007.

[9] J. Katz, J.Shin, Parallel and Concurrent Security of the HB and HB$^+$ Protocols, Cryptology ePrint Archive, Report 2005/461, 2005, http://eprint.iacr.org.

[10] J. Katz and A. Smith. Analysing the HB and HB$^+$ Protocols in the "Large Error" Case. Available from http://eprint.iacr.org/2006/326.pdf.

[11] J.Bringer, H.Chabanne, E, Dottax, HB^{++}: a Lightweight Authentication Protocol Secure against Some Attacks, IEEE International Conference on Pervasive Sevices, Workshop on Security, Privacy and Trust in Pervasive and Ubiquitous Computing - SecPerU, 2006.

[12] D.N. Duc, and K. Kim. Securing HB+ against GRS Man-in-the-Middle Attack, Proceedings of the Symposium on Cryptography and Information Security (SCIS2007), 2007.

[13] H. Krawczyk. LFSR-based hashing and authentication. In Yvo Desmedt, editor, CRYPTO, volume 839 of Lecture Notes in Computer Science, pages 129C139. Springer, 1994.
[14] H. Krawczyk. New Hash Functions for Message Authentication. In Proceedings of Eurocrypt 1995, LNCS 950, pp. 301C310, Springer, 1995.